Event Processing Track Summary for CHEP 2012

Adam L. Lyon

Scientific Computing Division
Fermi National Accelerator Laboratory
PO Box 500, Batavia, IL 60510-0500 USA

lyon@fnal.gov

Abstract. A general perspective of the CHEP 2012 Event Processing track is given, and some predictions for the future are offered.

1. Introduction

The event processing track includes a broad set of topics regarding the process by which events in high energy physics detectors are converted from raw detector signals into physics analyses and finally a set of results. There were 84 contributions to this track, 20 talks and 64 posters, on the subjects of tracking algorithms, calibration and alignment, simulation, event displays, analysis frameworks, ROOT, and the use of multicore and graphics processors. Every contribution should have a paper in these proceedings, so I will not attempt to summarize any of them individually. Rather, I will give an overall sense of perspective of the challenges and solutions discussed and some predictions for the future for several topics. I will try not to explicitly refer to individual talks and posters.

2. Processing on multicores is becoming a necessity

Multicore CPUs have been standard for the last several years and the number of cores per processor has been steadily increasing. Purchasing machines with 16, 32 or even 64 cores is routine. Multicore nodes are typically utilized by event based parallel processing. That is each core processes an event at a time. In this way, a node with 64 cores can process 64 events simultaneously. The memory on the node necessary to support that parallel processing also scales up with the number of cores. Fortunately, the price of memory continues to fall such that 2 GB/core is usually attainable. However, a new problem beginning to arise is that 2 GB/core is no longer sufficient to process one event, especially high multiplicity LHC events due to pileup.

The emerging solution is to parallelize the event processing algorithms themselves. An example could be to perform hit clustering such that different regions of the detector are handled by different cores running simultaneously. In this way the memory requirements are spread amongst the cores. Of course event processing has many steps that must occur serially. One cannot perform track finding without first clustering the hits. These dependencies make parallelization difficult, but there are several frameworks under investigation by many experiments that can help (e.g. libDispatch[1] and Threading Building Blocks[2]).

While the event boundaries are of course crucial for understanding physics, perhaps it is time to blur this boundary for optimal use of multiple cores. Typically, processing occurs on an event-by-event basis where the application does not move on to the next event until the current event is completely processed (e.g. all hits clustered, all tracks found, etc). Even if the algorithms are deeply

parallelized, one must always wait for the longer running threads to finish before moving to the next event. This wait can be sizable. One solution is to rethink the event boundary and perhaps have the application begin processing pieces of other events even before the current event is finished. This, of course, makes processing much more complicated and perhaps difficult to think about, but the gains in overall speed may be worth this effort.

Subtle details of i/o management can cause problems for efficient use of multiple cores. For example, ROOT[3] typically groups events into "baskets" which may be compressed. If two threads are reading events that reside in the same basket, each thread may perform the decompression, thereby duplicating effort unnecessarily. Careful optimization of i/o is necessary and is under investigation.

A prediction for CHEP 2013 is that deep parallelization of algorithms will be a hot topic and we will see many clever ideas for simultaneously running as many parts of event processing as possible.

3. Graphics processing units are making the impossible possible

For suitable algorithms, GPUs offer astounding speed improvements (100x, 200x) compared to conventional CPUs. Use of GPUs in HEP computing was first mentioned at CHEP 2010 and has clearly attracted enormous interest since then. For example, several experiments, including PANDA and CBM, are investigating software-only trigger systems to do real time reconstruction and 1000x rate reduction. Such a system is only feasible economically with the speed of a GPU farm. Many algorithms are being modified to be parallelized within GPUs with impressive speed improvements.

GPUs, however, are not suitable for every algorithm or processing task. Therefore one needs a hybrid system to run processing jobs on conventional CPU farms and GPU farms, using the one that is the most efficient for particular tasks. A workflow system is required to make this processing seamless, and the Condor team is investigating integrating GPU farms into their job management system.

Certainly the speed improvements of GPUs are incredibly tantalizing and solutions for integrating them into HEP computing are beginning to emerge. At CHEP 2013, one would expect to see much progress on these fronts.

4. Common frameworks are becoming more common

An experiment's framework supplies services for the more mundane lower-level event processing tasks, like i/o, shared object loading, persistency, enforcing the event data model, dispatching, handling links between physics objects, etc. Such a system allows physicists to concentrate on algorithms and results instead of lower-level computer software. The large experiments typically have resources and interested people to write their own frameworks. It can take a dedicated group to really do this right due to the many subtle details that need to be worked out. Smaller experiments, on the other hand, may not have the resources to write a full-fledged framework. Since such experiments are becoming more prevalent at laboratories such as Fermilab and FAIR, the labs themselves, in close collaboration with their experiments, are writing common framework systems meant to be used by many experiments. FairRoot[4] from GSI is a ROOT based common framework that was first mentioned at CHEP 2007 and provides a broad and common set of services that are used as a base for an experiment's specific framework. New for this conference, Art[5] from Fermilab is a lite, forked version of the CMS framework and is being used by the Intensity Frontier experiments like NOvA, Mu2e, g-2, etc.

Writing these frameworks offers the labs' computing divisions close collaboration with their experiments. One expects in CHEP 2013 to see more experiments signing on to such frameworks, especially as they realize it requires special expertise to write a successful system.

5. Accurate and efficient simulations are crucial

Validation of the models, processes and physics lists within Geant4[6] has been a recurring theme at nearly every CHEP. There is a large industry comparing such models to experimental data as well as to each other in regions where they overlap. Along with improvements in the models themselves,

much effort has gone into organizing the comparison information and making such data easy to find and view (see https://g4validation.fnal.gov:8080/G4HadronicValidation).

In Geant4, as in any large codebase that has grown over time, there are opportunities for refactoring. Such work is not easy, but the benefits of improved accuracy and performance can outweigh the cost. There has been work to identify areas where refactoring may produce some benefits, such as reducing coupling, eliminating duplicated code, and speeding up algorithms. It has been noted that many models that are deemed to be slow are so only because of an inefficient component or algorithm that can perhaps be excised or fixed. It was also noted that one could consider replacing some algorithms with experimental data.

For CHEP 2013, it is clear we will see more progress on these fronts and perhaps news of Geant5.

6. Sophisticated detectors require sophisticated algorithms

The event processing track is a great place to present clever and new ideas in event reconstruction algorithms, and this CHEP did not disappoint. An especially interesting idea is to borrow algorithms from the medical imaging industry for vertex finding in high multiplicity events (see ref. 7). It makes little sense to attempt to simplify the ideas and algorithms in the track here when they are fully explained elsewhere in these proceedings, so the reader is encouraged to peruse the contributions.

7. Summary of the event processing track

There were excellent talks and posters on a broad range of topics, indicating that enormous efforts are occurring in many areas of HEP computing. At CHEP 2013, we should expect to see use of multicore and GPUs mature. Deep parallelization of event processing algorithms will be a hot topic. Smaller experiments will hopefully be able to take advantage of such improvements as the shared frameworks adopt these techniques as well. The continual validation of GEANT4 will become easier with the organizational improvements being implemented. The sophistication of algorithms will continue to increase, but one hopes that the complexity will not, and perhaps looking to other fields will make that goal easier.

Unlike previous CHEPs where there was large emphasis on getting the initial set of software and algorithms working, especially for the LHC experiments, at this CHEP the general tone was "In general our software works well, but we need to make it go faster!" All of the investigations into multicore and GPU computing are with that goal of speed in mind. When such techniques come to fruition, one can expect to process very complicated events seen by very sophisticated detectors without an enormous increase in computing capacity that would otherwise be necessary to keep up.

References
[1]   http://libdispatch.macosforge.org/
[2]   http://threadingbuildingblocks.org/
[3]   R. Brun and F. Rademakers, "ROOT – An Object Oriented Data Analysis Framework", in Proceedings AIHENP'96 Workshop, Lausanne, Sep. 1996, Nucl. Inst. & Meth. In Phys. Res. A 389 (1997) 81-86. See also http://root.cern.ch/
[4]   http://fairroot.gsi.de/
[5]   W. Brown, et. al., "The art framework", in these proceedings

[6]   S. Agostinelli, et. al., Nucl. Inst. & Meth. In Phys. Res. A 506 (2003) 250; J. Allison, et. al., IEEE Transactions on Nuclear Science 53 (2006) 270; http://geant4.cern.ch/

[7]   S.G. Hageboeck, et. al., "Medical imaging inspired vertex reconstruction at the large hadron collider", in these proceedings